

## Report. z5150006 (Revision)

1.

It implements the 3-way handshake and 4-wave hand as well as implement a reliable transmission according to fast retransmission and retransmission due to timeout both packets are transfer through the PLD module including the retransmission packets. Moreover window slide achieved and also set a reused buffer both on sender and receiver side to alleviate the memory usage with the concept of memory replacement. According to the pipelined inflight packets transmission the performance might be optimal.

2.

Packet Type	Seq Num	Ack Num	Payload Size	Checksum	Window Size	Number of packets	Max Segment Size
1 Byte	4 Bytes	4 Bytes	4 Bytes	8 Bytes	4 Bytes	4 Bytes	4 Bytes

Total size of the segment header is 33 Bytes.

It includes the type of the packet like "SYN" or "ACK" or "FIN".

And the sequence number as well as ack number, payload size the checksum for corrupt packet detection and window size, number of packets and the size of max segment for the use of buffer reuse management on the receiver side.

3.

- My program currently implement the slide window to increase the inflight packets on the sender side and using vector/buffer to cache the packet in byte[], if retransmission needed will pick up the data from the vector.
- On the receiver side also implement a buffer to cache the received data and when it is in the maximum correct sequence will flush them out to the copy file.
- The problem is still not quite sure if send the data in window at one time (which means in one loop to send them out) or just let the one packet sending out one time each loop. The program still need to improve for the design and for the efficiency and robust.

4.

The core part of snippet referenced from the web forum as below.

If using Java 1.6+:

```
1. arrayToResize = Arrays.copyOf(arrayToResize, arrayToResize.length * 2);
```

Also reference some common question from the forum and Google website.

5.

(a) Run your protocol using pDrop = 0.1, MWS = 500 bytes, MSS = 100 bytes, seed = 100, gamma = 4, and pDuplicate, pCorrupt, pOrder, MaxOrder, pDelay, MaxDelay all set to 0. Transfer the file **test0.pdf** (available on the assignment webpage). The file should be received correctly at the Receiver. Show the sequence of STP packets that are observed at the Receiver. It is sufficient to just indicate the sequence numbers of the STP packets that have arrived. Run an additional experiment with pdrop = 0.3, transferring the same file (**test0.pdf**). In your report, discuss the resulting packet sequences of both experiments indicating where dropping occurred. Also, in the appendix section show the packet sequences for both the experiments.

**pDrop = 0.1:**

drop	0.37	D	601	100	1
rcv	0.37	A	1	0	601
snd	0.37	D	701	100	1
rcv/DA	0.38	A	1	0	601
snd	0.38	D	801	100	1
rcv/DA	0.38	A	1	0	601
snd	0.38	D	901	100	1
rcv/DA	0.40	A	1	0	601
snd	0.40	D	1001	100	1
snd/RXT	0.40	D	601	100	1
rcv/DA	0.40	A	1	0	601

Obviously we can see the drop happened at sequence number 601 then followed by 3 duplicated ACKs leading to a fast retransmission.

**pDrop = 0.3:**

drop	1.92	D	601	100	1
snd/RXT	1.92	D	501	100	1
rcv	1.93	A	1	0	601
snd	1.93	D	701	100	1
rcv/DA	1.93	A	1	0	601
snd	1.93	D	801	100	1
rcv/DA	1.93	A	1	0	601
snd	1.93	D	901	100	1
rcv/DA	1.95	A	1	0	601
snd	1.95	D	1001	100	1
drop	1.95	D	601	100	1
rcv/DA	1.95	A	1	0	601
snd/RXT	2.96	D	601	100	1

The similar situation also found when set the pDrop = 0.3, more packets dropped due to the increasing of probability of drop event. Also found 3 duplicate ACKs at Seq 601 for the fast retransmission and in the snapshot another retransmit may result from retransmission due to timeout.

**(b)** The timeout for STP is given by:

$$\text{TimeoutInterval} = \text{EstimatedRTT} + \gamma * \text{DevRTT}$$

where gamma will be supplied to the program as an input argument, see Section 4.5.

Set pdrop = 0.5, MWS = 500 bytes, MSS = 50 bytes, seed = 300, pdelay = 0.2, MaxDelay = 1000 and pDuplicate, pCorrupt, pOrder, MaxOrder all set to 0. Run three experiments with the following different gamma values:

- gamma = 2
- gamma = 4
- gamma = 6

and transfer the file **test1.pdf** using STP. Show a table that indicates how many STP packets were transmitted in total and how long the overall transfer took. Discuss the results.

Gamma = 2	Gamma = 4	Gamma = 6
17828 (total segments)	18887 (total segments)	17822 (total segments)
3 mins	2.16 mins	2.9 mins

#### ➤ Gamma=2

```

drop      183.37      D      308001      50      1
snd      183.37      D      308151      50      1
rcv/DA    183.38      A           1      0      308001
snd/RXT   183.38      D      308001      50      1
snd/dely  183.38      D      308201      3      1
snd/RXT   183.38      D      308001      50      1
rcv      183.38      A           1      0      308101
snd/RXT   183.38      D      308101      50      1
rcv      183.38      A           1      0      308201
snd/RXT   183.38      D      308201      3      1
snd      183.39      F      308204      0      1
rcv      183.39      A           1      0      308204
rcv      183.39      A           1      0      308204
=====
Size of the file (in Bytes)                      308203
Segments transmitted (including drop & RXT)      17828
Number of Segments handled by PLD                17825
Number of Segments dropped                       8851
Number of Segments Corrupted                    0
Number of Segments Re-ordered                   0
Number of Segments Duplicated                   0
Number of Segments Delayed                      1786
Number of Retransmissions due to TIMEOUT        230
Number of FAST RETRANSMISSION                  1453
Number of DUP ACKS received                     4359
=====

```

#### ➤ Gamma=4

```

drop      129.80      D      308201      3      1
snd/dely  129.86      D      308201      3      1
snd/RXT   129.92      D      308201      3      1
rcv/DA    129.92      A           1      0      308201
snd/RXT   129.92      D      308201      3      1
drop      129.94      D      308201      3      1
rcv/DA    129.94      A           1      0      308201
drop      129.94      D      308201      3      1
snd      129.94      F      308204      0      1
rcv      129.94      A           1      0      308204
rcv      129.94      F           1      0      308204
snd      129.94      A      308205      0      2
=====
Size of the file (in Bytes)                      308203
Segments transmitted (including drop & RXT)      18887
Number of Segments handled by PLD                18883
Number of Segments dropped                       9379
Number of Segments Corrupted                    0
Number of Segments Re-ordered                   0
Number of Segments Duplicated                   0
Number of Segments Delayed                      1902
Number of Retransmissions due to TIMEOUT        1321
Number of FAST RETRANSMISSION                  1567
Number of DUP ACKS received                     4703
=====

```

#### ➤ Gamma=6

snd/RXT	175.91	D	308201	3	1
snd	175.92	F	308204	0	1
rcv	175.92	A	1	0	308204
rcv	175.92	F	1	0	308204
snd	175.92	A	308205	0	2
=====					
Size of the file (in Bytes)					308203
Segments transmitted (including drop & RXT)					17822
Number of Segments handled by PLD					17818
Number of Segments dropped					8850
Number of Segments Corrupted					0
Number of Segments Re-ordered					0
Number of Segments Duplicated					0
Number of Segments Delayed					1785
Number of Retransmissions due to TIMEOUT					197
Number of FAST RETRANSMISSION					1492
Number of DUP ACKS received					4478
=====					

From the table and snapshot above we can conclude that with the increasing of Gamma the overall transfer consuming almost equal time and according to the formula  $\text{TimeoutInterval} = \text{EstimatedRTT} + \text{gamma} * \text{DevRTT}$ . The weigh of DevRTT dominate the TimeoutInterval which presents **gamma=4** is the best value for the timeout interval calculation so that the transfer has the best performance.

However maybe at a certain time if the more data packets transmitted the less time will be consumed which assumed from above statistics.

(c) Use the following values and run STP to transfer test2.pdf.

MWS=500bytes MSS=50 gamma=4 pDrop=0.1 pDuplicate=0.1 pCorrupt=0.1

pOrder=0.1 maxOrder=4 pDelay=0 maxDelay=0 seed=300

Has the file been successfully transferred? How long the overall transfer took? For this experiment, which of the factor (out of pDrop, pDuplicate, pCorrupt and pOrder) is the most critical contributing most in the overall transfer time? How have you determined this?

Provide the screen shot for the initial transfer (connection establishment + first 20 entries) and the last 20 entries plus the summary statistics table for the sender\_log.txt and receiver\_log.txt files in appendix. Do not attach the complete log files due to their sizes.

Yes. It is transmitted successfully. It totally took **16.2** mins to finish the transferring.

From my perspective of view, the most critical condition is the number of transited packets (i.e. a relation between file size and max segment size). Because comparing with other experiments the other factors are the same (e.g. the probability of drop, re-order, duplicate, gamma, window size, etc.) but only the file size has dramatically changed... Moreover the weigh of other arguments have the same value 0.1. So we can have this conclusion properly. As the screenshots show that the number of packets reached to **107046** segments.

## Appendix

The First 20 log records.

snd	0.00	S	0	0	0
rcv	0.05	SA	0	0	1
snd	0.05	A	1	0	1
snd	0.05	D	1	50	1
rcv	0.05	A	1	0	51
snd	0.05	D	51	50	1
rcv	0.05	A	1	0	101
snd	0.05	D	101	50	1
rcv	0.06	A	1	0	151
snd	0.06	D	151	50	1
rcv	0.06	A	1	0	201
snd/dup	0.06	D	201	50	1
rcv	0.06	A	1	0	251
snd/dup	0.06	D	251	50	1
rcv	0.06	A	1	0	301
snd	0.06	D	301	50	1
rcv	0.06	A	1	0	351
drop	0.06	D	351	50	1
snd/RXT	0.13	D	351	50	1
snd	0.13	D	401	50	1
rcv	0.13	A	1	0	401
snd/RXT	0.13	D	401	50	1
snd/dup	0.13	D	451	50	1
rcv	0.13	A	1	0	451
snd/RXT	0.13	D	451	50	1
snd	0.13	D	501	50	1
rcv	0.13	A	1	0	501
drop	0.13	D	501	50	1
snd	0.13	D	551	50	1

The Last 20 log records.

drop	976.16	D	1605501	50	4194305
snd/RXT	976.22	D	1605501	50	4194305
snd/dup	976.22	D	1605501	50	4194305
snd/RXT	976.28	D	1605501	50	4194305
snd/RXT	976.34	D	1605501	50	4194305
snd/RXT	976.40	D	1605501	50	4194305
snd/RXT	976.46	D	1605501	50	4194305
snd/RXT	976.53	D	1605501	50	4194305
snd/RXT	976.59	D	1605501	50	4194305
snd/RXT	976.65	D	1605501	50	4194305
snd/corr	976.71	D	1605501	50	4194305
snd	976.71	F	1605586	0	1
rcv	976.71	A	1	0	1605586
rcv	976.71	F	1	0	1605586
snd	976.71	A	1605587	0	2
=====					
Size of the file (in Bytes)					1605585
Segments transmitted (including drop & RXT)					107046
Number of Segments handled by PLD					107042
Number of Segments dropped					10853
Number of Segments Corrupted					8712
Number of Segments Re-ordered					6239
Number of Segments Duplicated					9577
Number of Segments Delayed					0
Number of Retransmissions due to TIMEOUT					3933
Number of FAST RETRANSMISSION					10353
Number of DUP ACKS received					31059
=====					